**SI 206 Final Project Report**

**Arya Pudota, Nilay Thummar**

**Project Goals**

The goals of this project are to create a tool that allows us to conduct analysis on the NCAA College Football sport. We are aiming to look into understanding if there is a correlation between a team’s AP Poll ranking and the number of news articles about them in a given week, and also understand the total amount of media attention the NCAA football season receives over all the weeks.

The goal was to create this project using the information gained from the SI 206 course. We aimed to use Python3, with SQLite3, Matplotlib and other associated libraries to build our application.

The visualizations that we aimed to create was a bar graph that compared the number of articles for a given ranking in a week added up for the season and looked for a positive correlation between a higher rank and more news articles, and another visualization that looked at the total number of articles of the Top 25 teams for every week, looking to see patterns for which weeks were under and over reported by the media.

**Achieved Goals**

We were able to achieve the goals that we set in place for ourselves, we were able to find two APIs that allowed us to find the data that we needed. This was the CollegeFootballData API and the ContextualWeb API.

We were also able to create the visualizations that we wanted, the first being a histogram that looked at the top 10 teams in terms of media coverage over the season, and the second being a bar graph that looked at overall media trends over the 15 weeks of the NCAA football season.

**Problems That We Faced**

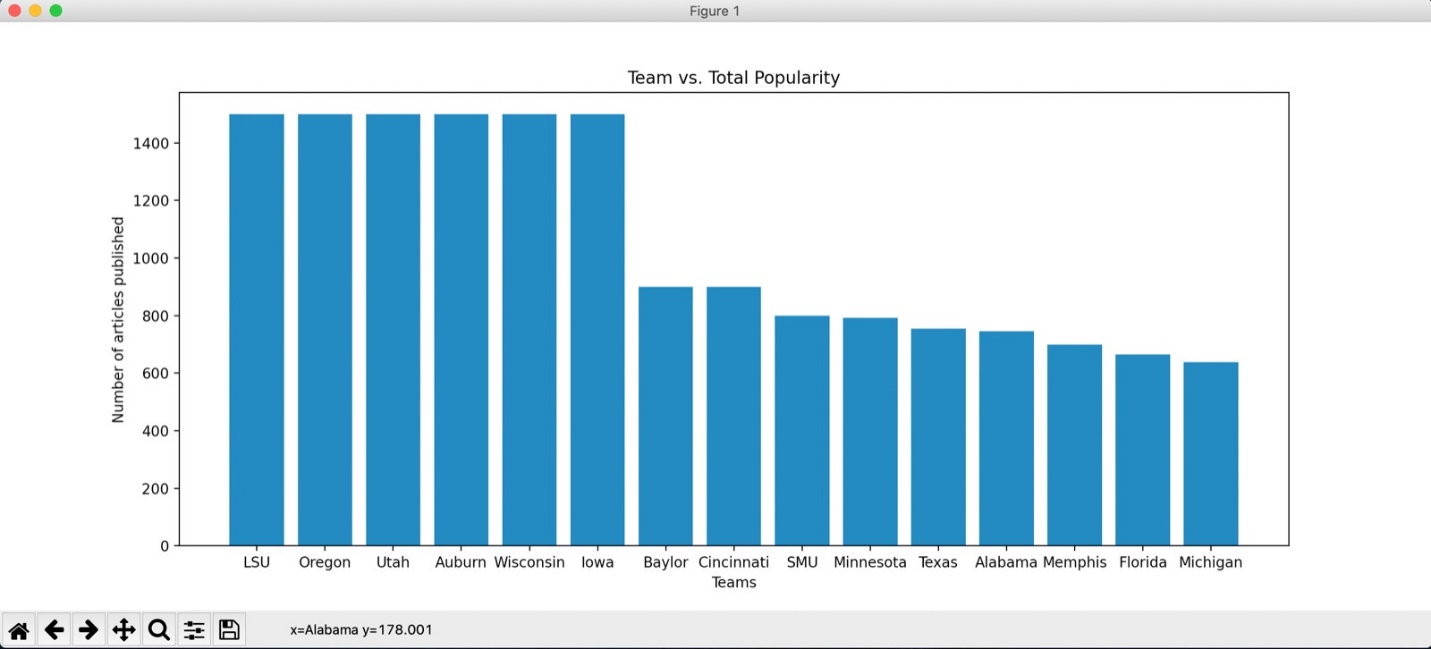
Finding the right set of APIs to use for our project was one of our concerns. It took us some time to find APIs for college football data that were free to access and contained the data that we wanted. We were finally able to find a website that met our needs, CollegeFootballData ([www.collegefootballdata.com](http://www.collegefootballdata.com))

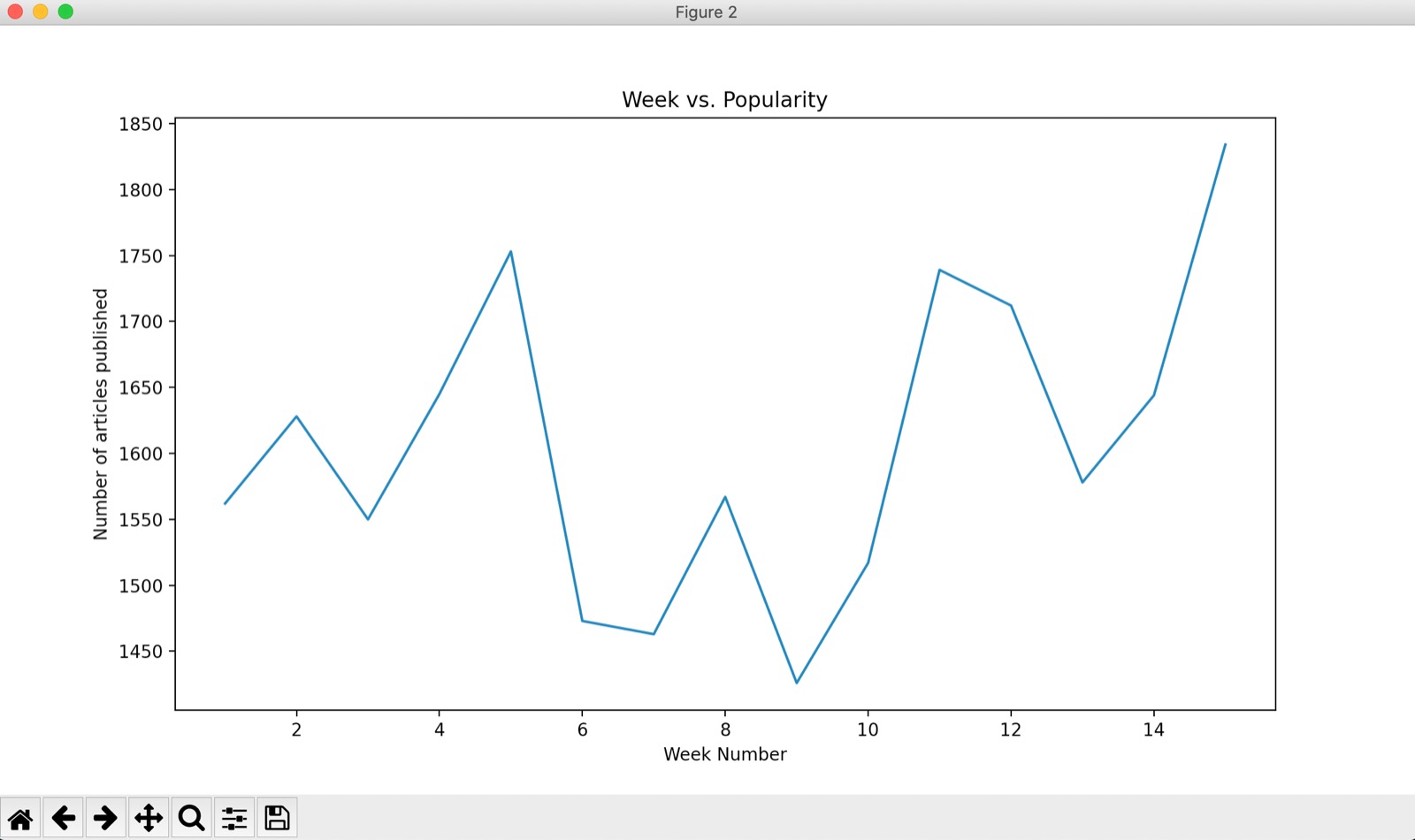
Another concern for us was with regards to handling duplicate values in our table. This was because we would need to try and find a way to enter in data for a team in our table that allowed for us to have multiple entries for the same team but for different weeks and years.

We were able to overcome this challenge by creating a unique key for each entry in the Team\_Data and Team\_News tables that allowed us to create aa distinct key for a specific team in a specific week in a specific year.

The last, and major, problem that we faced was with regards to our News API. Initially we were going to use the API provided by newsapi.org but we later realized that it doesn’t allow us to make queries for news that is older than 1 month. This was bad news for us as we were using data that was more than 6 months old in some cases. Hence we had to switch to a new API and we were able to find one that suited our needs in the form of the ContextualWeb API.

**Visualizations**





**Instructions For Running The Code**

Running our code is a fairly straightforward process. The first file to run is the retrieve.py function that retrieves all of the information required for the databases to be populated. This file can be run multiple times without duplicating the data as well.

The second file that we run is the visualize.py file that completes the calculations and the visualizations created by our program.

**Function Documentation**

**Functions for Information Retrieval**

1. **team\_rank\_information\_retrieval()**

**Inputs taken:** None

**Output:** None

**What this function does:**

This function’s purpose is to retrieve information from the CollegeFootballData API and store it in two of our SQL tables, the Team\_Data table and Team\_News table. It does this by querying the CollegeFootballData API with a request for the Top 25 teams for each week of the 2019 NCAA College Football season.

After this, it takes this data and stores it in our tables:

Team\_Data table contains the *teamrank\_id, year\_week, date, team\_id* and *ranking.*

Team\_Name table contains the *team\_id* and the *team\_name*

The purpose of the Team\_Data table is to store all of the information of the rankings of the different teams for every week in the season. The Team\_Name table is used to help interface the Team\_Data table with the Team\_News table from our other function

1. **team\_news\_information\_retrieval()**

**Inputs taken:** None

**Output:** None

**What this function does:**

This function’s purpose is to retrieve information from the ContextualWeb API about the frequency and popularity of the different teams in the NCAA Top 25 across the different weeks in the 2019 season. It does this by querying the ContextualWeb API for the specific team and the specific time period for the given week and retrieves the number of matching articles for the query.

After this, it takes this data and stores it in our table:

Team\_News table contains information about the *teamrank\_id, team\_id* and *num\_articles*

1. **get\_num\_news(teamname: str, date\_from:str, date\_to:str)**

**Inputs Taken:** teamname (string type), date\_from (string type), date\_to (string type)

**Output:** Outputs the number of articles matching the team for the given time period

**What this function does:**

This function is designed to be a helper function for the team\_news\_information\_retrieval() function. It helps by acting as a helper function that retrieves the number of articles published for a specific team within the time period that we specify.

**Functions for Calculations & Visualizations**

1. **create\_viz\_1()**

**Input: None**

**Output:** Outputs a histogram that details the relationship between a given team in the NCAA football championship and its news coverage. We compare the number of articles curated over the entire season for the given teams and compare the top 25 teams by number of news articles.

1. **create\_viz\_2()**

**Input: None**

**Output:** This function outputs a line graph of the trends seen in terms of total news articles published about the top 25 teams in the NCAA Football championship. Looking at the graph we can see a positive correlation between going through towards the end of the season and growth in the number of articles.

**Resources Used**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Issue Description** | **Location of Resource** | **Result (did it solve the issue?)** |
| 4/25/2020 | Finding an API that would allow us to source college football ranking data effectively | api.collegefootballdata.com | We were able to solve this issue as the API we found was able to give us the information we needed in order to carry out our visualizations |
| 4/25/2020 | Finding an API that would allow us to source news articles effectively | newsapi.org | This did not solve the issue as we were unable to retrieve article information that was more than 30 days old without using a paid subscription plan |
| 4/26/2020 | Finding an API that would allow us to source news articles effectively | ContextualWeb API | This API was able to give us the information that we needed about the teams and find articles dating back the the months of August-December 2019 |
| 4/27/2020 | Help with matplotlib due to issues with syntax | W3Schools | We were able to debug our code and find the syntactical errors associated with our code |
| 4/27/2020 | Finding a way to store information for the second visualization | Numpy | We were able to use a Numpy array to deal with the issues we were facing in terms of performance and ability to sort with the existing data structures that we were using. We also made use of the W3Schools resource to learn more about Numpy. |